

II. AMENDMENTS TO THE CLAIMS

The following listing of claims is a courtesy copy only, as no claims have been amended by this response:

1. (Original) A method of designing an integrated circuit (IC) design, the method comprising the steps of:

 determining net shapes for a net;

 sorting edges of the net shapes in preparation for a plane sweep into a sorted list;

 using a plane sweep algorithm to detect an intersection of at least two shapes;

 creating a new shape from the intersecting shapes in the case that the intersection is not totally contained by one of the intersecting shapes, and adding the new shape to an active list of the plane sweep;

 adding the new shape to a shapes list for the net in the case that the new shape requires a larger spacing than the intersecting shapes; and

 routing the IC design and including the new shape on a shapes list for the net as a blockage to prevent spacing errors during routing.

2. (Original) The method of claim 1, further comprising the step of checking the design for design rule violations.

3. (Original) The method of claim 2, further comprising the step of including shapes on the shape list as shapes on the net during the checking step.

4. (Original) The method of claim 1, further comprising the step of removing the new shape as a blockage after the routing step.
5. (Original) The method of claim 1, wherein the creating step includes assigning edges of the new shape to be distal edges of the intersecting shapes.
6. (Original) The method of claim 1, wherein the creating step further includes adding a high edge of the new shape into the sorted list.
7. (Original) The method of claim 1, wherein the using, creating and adding steps are repeated until all edges on the sorted list have been evaluated.
8. (Original) A system for designing an integrated circuit (IC) design, the system comprising:
 - means for determining net shapes for a net;
 - means for sorting edges of the net shapes in preparation for a plane sweep into a sorted list;
 - means for using a plane sweep algorithm to detect an intersection of at least two shapes;
 - means for creating a new shape from the intersecting shapes in the case that the intersection is not totally contained by one of the intersecting shapes, and adding the new shape to an active list of the plane sweep;
 - means for adding the new shape to a shapes list for the net in the case that the new shape

requires a larger spacing than the intersecting shapes; and
means for routing the IC design and including the new shape on a shapes list for the net
as a blockage to prevent spacing errors during routing.

9. (Original) The system of claim 8, further comprising means for checking the design for
design rule violations.

10. (Original) The system of claim 9, wherein the checking means further includes shapes on the
shape list as shapes on the net during the checking step.

11. (Original) The system of claim 8, wherein the routing means further removes the new shape
as a blockage after the routing step.

12. (Original) The system of claim 8, wherein the creating means assigns edges of the new
shape to be distal edges of the intersecting shapes.

13. (Original) The system of claim 8, wherein the creating means further adds a high edge of the
new shape into the sorted list.

14. (Original) The system of claim 8, wherein the using, creating and adding means repeat their
functions until all edges on the sorted list have been evaluated.

15. (Original) A computer program product comprising a computer useable medium having computer

readable program code embodied therein for designing an integrated circuit (IC) design, the program product comprising:

program code configured to determine net shapes for a net;

program code configured to sort edges of the net shapes in preparation for a plane sweep into a sorted list;

program code configured to use a plane sweep algorithm to detect an intersection of at least two shapes;

program code configured to create a new shape from the intersecting shapes in the case that the intersection is not totally contained by one of the intersecting shapes, and adding the new shape to an active list of the plane sweep;

program code configured to add the new shape to a shapes list for the net in the case that the new shape requires a larger spacing than the intersecting shapes; and

program code configured to route the IC design and including the new shape on a shapes list for the net as a blockage to prevent spacing errors during routing.

16. (Original) The program product of claim 15, further comprising program code configured to check the design for design rule violations.

17. (Original) The program product of claim 16, wherein the checking code further includes shapes on the shape list as shapes on the net during the checking step.

18. (Original) The program product of claim 15, wherein the routing code further removes the new shape as a blockage after the routing step.

19. (Original) The program product of claim 15, wherein the creating code assigns edges of the new shape to be distal edges of the intersecting shapes.

20. (Original) The program product of claim 15, wherein the creating code further adds a high edge of the new shape into the sorted list.